

SNRO Kiosk Printers
Software Development Kit
Reference Manual
Ver: 3.0

Shenzhen Xinyou Technology Co.,td.
Software research and development department

Revision history				
Version	explain	author	Approval	Revision date
3.0	Create			2020-06-01

Catalog

1. Brief Introduction	7
2. Applicable Models	7
3. Print interface description	7
3.1 SetCommands Mode(SetCommmandmode)	7
3.2 Clear The Buffer(SetClean).....	8
3.3 Set Line Space(SetLinespace)	8
3.4 Set Character Space(SetSpacechar).....	8
3.5 Set Chinese Space(SetSpacechinese)	9
3.6 Set Left Margin(SetLeftmargin).....	9
3.7 SetCutting blackmark offset(SetMarkoffsetcut)	9
3.8 Set Feed blackmark offset(SetMarkoffsetprint).....	10
3.9 Set Chinese Size(SetSizechinese).....	10
3.10 Set Character Size(SetSizechar).....	11
3.11 Set Text Size(SetSizetext)	11
3.12 Set Character Alignment(SetAlignment)	11
3.13 Set Bold Character(SetBold)	12
3.14 Set Character Rotation(SetRotate).....	12
3.15 Set Character Direction(SetDirection)	13
3.16 Set Character White(SetWhitemodel).....	13
3.17 Set Character Italic (SetItalic)	13

3.18	Set Character Underline(SetUnderline).....	14
3.19	Set Chinese Character Mode (SetReadZKmode).....	14
3.20	SetHorizontal Tab (SetHTseat).....	14
3.21	Set National font and Code page(SetCodepage).....	15
3.22	Set NV bit Image(SetNvbmp).....	17
3.23	Set Right margin(SetRightmargin)	18
3.24	Set 1D Bar Code Alignment (Set1DBarCodeAlign)	18
3.25	Print Self-test Page (PrintSelfcheck).....	18
3.26	Print Feed Line(PrintFeedline)	19
3.27	Print Character String(PrintString).....	19
3.28	Print And Change Lines (PrintChargeRow).....	20
3.29	Paper Feed Dot (PrintFeedDot).....	20
3.30	PrintNext Level Tab (PrintNextHT).....	20
3.31	Print Cut paper (PrintCutpaper).....	21
3.32	Black mark detection (PrintMarkposition).....	21
3.33	Feed blackmark to printing position(PrintMarkpositionPrint)	21
3.34	Feed blackmark to cut position (PrintMarkpositioncut)	22
3.35	Print & cut on BM paper(PrintMarkcutpaper)	22
3.36	Print QR code(PrintQrcode).....	22
3.37	Print QR code for T500II(PrintQrCodeT500II)	23
3.38	Print PDF417 code (PrintPdf417)	24

3.39	Print 1D code(Print1Dbar)	24
3.40	Print Disk BMP file(PrintDiskbmpfile).....	25
3.41	Black And White Print BMP/JPG/PNG image files (PrintDiskImagefile)	25
3.42	Print BMP/JPG/PNG image files (PrintDiskImagefile).....	26
3.43	Print NV BMP file(PrintNvbmp)	26
3.44	Get Printer Status(GetStatus).....	26
3.45	Get SDK Information(GetSDKInformation).....	28
3.46	Converts string to an array(JNASTringToByte).....	28
3.47	Converts array to an string (JNAByteToString).....	28
3.48	Converts array to an string(JNAByteToString).....	29
4.	USB connection print.(UsbDirver.java)	29
4.1	To adapt to the development environment	29
4.2	ConnectMethods.....	29
4.2.1	Print byte[]Array1 (write)	29
4.2.2	Print byte[]Array2 (write)	29
4.2.3	Print byte[]Array3(write)	30
4.2.4	Print byte[]Array4(write).....	30
4.2.5	Read Data1 (read)	31
4.2.6	Read Data2 (read)	31
4.2.7	USB Permission Status (isUsbPermission)	32
4.2.8	Connection Status (isConnected)	32
4.2.9	Set timeout (setFlowCtrl)	32
4.2.10	Set PendingIntent(setPermissionIntent).....	33
4.2.11	Connect Printer Intent1 (usbAttached)	33
4.2.12	Connect Printer Intent2 (usbAttached)	33
4.2.13	Remove Connection Intent (usbDetached)	34
4.2.14	Open UsbDevice1 (openUsbDevice)	34
4.2.15	Open UsbDevice2 (openUsbDevice)	35
4.2.16	Close UsbDevice1 (closeUsbDevice)	35
4.2.17	Close UsbDevice2 (closeUsbDevice)	35
4.3	Sample.....	36
4.4	Matters Needing Attention.....	38

5.	Bluetooth Connection Print	39
5.1	To adapt to the development environment	39
5.2	ConnectMethods (BluetoothManager.java)	39
5.2.1	Connect Bluetooth Server(connectServer)	39
5.2.2	Send Data or Instruction (send)	39
5.2.3	Send Data or Instruction in batches (send)	40
5.2.4	Read Data (read).....	40
5.2.5	Shutdown Client (shutdownClient)	41
5.2.6	Whether The Socket Connection (isConnected)	41
5.3	Sample.....	41
6.	Serial Connection Print.....	42
6.1	To adapt to the development environment.....	42
6.2	Sample.....	42
6.2.1	Specify device name(SetDevname)	42
6.2.2	Init printer(SetInit).....	42
6.2.3	Close device(SetClose)	43
6.2.4	PrintData(PrintTransmitJNA).....	43
6.2.5	Read data(GetTransmitJNA).....	43
6.3	Sample.....	44
6.4	Matters needing attention.....	44

1. Brief Introduction

This reference manual, including the serial port, USB port, Bluetooth printer's print interface description; most of them are the same interface, the different parts of the interface will have a special note:

- The common print instruction function package ((printsdk-v2.0.jar))

To provide a common print operation instructions for the package, the most common interface instructions, the individual to adapt to specific models.

- USB Connect(usbprintsdk-v2.0.jar)
- Bluetooth SDK (btprintsdk-v2.0.jar)
- Serial port connect helper (SerialHelper.java)

2. Applicable Models

SNR-XXX/MS-XXXX full series

Serial port, USB and Bluetooth interface,are supported

3. Print interface description

3.1 SetCommands Mode(SetCommmandmode)

[Function]

Set printer commands mode

[Parameters]

int iMode: 2 EPIC Mode、 3 EPOS Mode

[Returned Value]

byte[] array

3.2 Clear The Buffer(SetClean)

[Function]

Clear buffer, clear previous setting parameters

[Parameters]

None

[Returned Value]

byte[] array

3.3 Set Line Space(SetLinespace)

[Function]

Set line space

[Parameters]

int iLinespace: line spacing, value is 0-127, units is 0.125mm

[Returned Value]

byte[] array

3.4 Set Character Space(SetSpacechar)

[Function]

Set character space

[Parameters]

int iSpace: character pitch, value is 0-64, units is 0.125mm

[Returned Value]

byte[] array

3.5 Set Chinese Space(SetSpacechinese)

[Function]

Set Chinesespace

[Parameters]

int iChsleftspace: left of Chinese is empty, values is 0-64, units is 0.125mm

int iChsrightspace: right of Chinese is empty, value is 0-64,units is 0.125mm

[Returned Value]

byte[] array

3.6 Set Left Margin(SetLeftmargin)

[Function]

Set left margin

[Parameters]

int iLeftspace: character space, value is0-576, units is 0.125mm

[Returned Value]

byte[] array

3.7 SetCutting blackmark offset(SetMarkoffsetcut)

[Function]

Set cutting blackmark offset value

[Parameters]

int iOffset: Offset, value is 0-1600

[Returned Value]

byte[] array

3.8 Set Feed blackmark offset(SetMarkoffsetprint)

[Function]

Set feed blackmark offset value

[Parameters]

int iOffset: Offset, value is 0-1600

[Returned Value]

byte[] array

3.9 Set Chinese Size(SetSizechinese)

[Function]

Set Chinese size

[Parameters]

int iHeight: Double-height 0 invalid 1 valid

int iWidth: Double-width 0 invalid 1 valid

int iUnderline: Undefine 0 invalid 1 valid

int iChinesetype: chinese type 0 24*24 1 16*16

[Returned Value]

byte[] array

3.10 Set Character Size(SetSizechar)

[Function]

Set character size

[Parameters]

int iHeight: Double-height 0 invalid 1 valid

int iWidth: Double-width 0 invalid 1 valid

int iUnderline: Underline 0 invalid 1 valid

int iAsciitype: ASCII 0 12*24 1 9*17

[Returned Value]

byte[] array

3.11 Set Text Size(SetSizetext)

[Function]

Set text size

[Parameters]

int iWidth: enlarge width, value is (1-8)

int iHeight: enlarge height, value is (1-8)

[Returned Value]

byte[] array

3.12 Set Character Alignment(SetAlignment)

[Function]

Set character alignment

[Parameters]

int iAlignment: 0 left、 1 center、 2 right

[Returned Value]

byte[] array

3.13 Set Bold Character(SetBold)

[Function]

Set bold character

[Parameters]

int iBold: 0 No 1 yes

[Returned Value]

byte[] array

3.14 Set Character Rotation(SetRotate)

[Function]

Set character rotation

[Parameters]

int iRotate: 0Remove rotate、 1Clockwise rotate 90°

[Returned Value]

byte[] array

3.15 Set Character Direction(SetDirection)

[Function]

Set character direction

[Parameters]

int iDirection: 0 From left to right, 1 rotate 180°

[returned Value]

byte[] array

3.16 Set Character White(SetWhitemodel)

[Function]

Set character inverse white

[Parameters]

int iWhite: 0 cancel white; 1 white

[Returned Value]

byte[] array

3.17 Set Character Italic (SetItalic)

[Function]

Set character Italic

[Parameters]

int italic: 0 cancel italic; 1 italic

[Returned Value]

byte[] array

3.18 Set Character Underline(SetUnderline)

[Function]

Set character Underline

[Parameters]

int underline: 0 cancel underline 1 one point underline ; 2 two point underline

[Returned Value]

byte[] array

3.19 Set Chinese Character Mode (SetReadZKmode)

[Function]

Set Chinese character Mode

[Parameters]

int mode: 0 to enter the Chinese character mode; 1 exit the Chinese character mode

[Returned Value]

byte[] array

3.20 SetHorizontal Tab (SetHTseat)

[Function]

Set Horizontal Tab

[Parameters]

byte[] bHTseat: The location of the horizontal tab from small to large units of a ASCII

character can not be 0.

int iLength: Horizontal tab of the number of location data.

[Returned Value]

byte[] array

3.21 Set National font and Code page(SetCodepage)

[Function]

Set National font and Code page

[Parameters]

int country: National font

0	United States
1	France
2	Germany
3	Britain
4	Denmarkl
5	Sweden
6	Italy
7	Spain
8	Japan
9	Norway

10	DenmarkII
----	-----------

int CPnumber: Code page

0	PC437[American European standards]
1	PC737
2	PC775
3	PC850
4	PC852
5	PC855
6	PC857
7	PC858
8	PC860
9	PC862
10	PC863
11	PC864
12	PC865
13	PC866
14	PC1251
15	PC1252
16	PC1253

17	PC1254
18	PC1255
19	PC1256
20	PC1257
21	PC928
22	Hebrew old
23	IINTEL CHAR
24	Katakana
25	Special Symbol 00-1F
26	SPACE PAGE

[Returned Value]

byte[] array

3.22 Set NV bit Image(SetNvbmp)**[Function]**

Set NV bit image

[Parameters]

int iNums: Bit image quantity(single file support 64k(max); total files support 192K)

const char* strPath: image file path (If only file name then please use current route, if specify all routes then use the specified route) , use " ;" separate, quantities need be same with iNums parameters.

[Returned Value]

byte[] array

3.23 Set Right margin(SetRightmargin)**[Function]**

Get SDK version information

[Parameters]

int iRightspace: right margin , value is 0 - 255

[Returned Value]

byte[] array

3.24 Set 1D Bar Code Alignment (Set1DBarCodeAlign)**[Function]**

When printing the bar code, the bar code is aligned according to the optional value of the iAlign

[Parameters]

int iAlign: Align type, 0 left, 1 center, 2 right

[Returned Value]

byte[] array

3.25 Print Self-test Page (PrintSelfcheck)**[Function]**

Print Self-test page

[Parameters]

None

[Returned Value]

byte[] array

3.26 Print Feed Line(PrintFeedline)

[Function]

Paper feed

[Parameters]

int iLine: Paper feed lines

[Returned Value]

byte[] array

3.27 Print Character String(PrintString)

[Function]

Print character string

[Parameters]

String strData: Print character string content

int ilmme;Whether the newline instruction 0x0a:0 add a line directive 1 without newline
(instruction until the next newline is print)

[Returned Value]

byte[] array

3.28 Print And Change Lines (PrintChargeRow)

[Function]

Print and Change lines, Without printing the contents of the time to walk a blank line

[Parameters]

None

[Returned Value]

byte[] array

3.29 Paper Feed Dot (PrintFeedDot)

[Function]

Paper, units is 0.125mm

[Parameters]

int Lnumber: value is 0-250

[Returned Value]

byte[] array

3.30 PrintNext Level Tab (PrintNextHT)

[Function]

Execution to the Next Level Tab

[Parameters]

None

[Returned Value]

byte[] array

3.31 Print Cut paper (PrintCutpaper)

[Function]

Cut paper,Note: only the printer with a cut knife is supported

[Parameters]

int iMode: 0 Full cut、 1 Partial cut

[Returned Value]

byte[] array

3.32 Black mark detection (PrintMarkposition)

[Function]

On black mode, check flag and stop flag position

[Parameters]

None

[Returned Value]

byte[] array

3.33 Feed blackmark to printing position(PrintMarkpositionPrint)

[Function]

On black mode, check flag and print flag position

SetMarkoffsetprint The offset has a effect on him

[Parameters]

None

[Returned Value]

byte[] array

3.34 Feed blackmark to cut position (PrintMarkpositioncut)

[Function]

On black mode, check flag and cut flag position

SetMarkoffsetcut The offset has a effect on him

[Parameters]

None

[Returned Value]

byte[] array

3.35 Print & cut on BM paper(PrintMarkcutpaper)

[Function]

Cut blackmark paper

[Parameters]

int iMode: 0 Detect BM paper full cut、 1 No detect BM paper partial cut

[Returned Value]

byte[] array

3.36 Print QR code(PrintQrcode)

[Function]

Print QR code

[Parameters]

String strData: Content

int iLmargin: left margin, value is 0-27 units mm

int iMside: Units length,, that is QR code size, value is 1-8, (some printers only support 1-4)

int iRound: Surrounded mode,0 print immediately (No mixed row) 、 1 surround (mixed row, some printers don' t support)

[Returned Value]

byte[] array

If can not print, please switch to PrintQrcode51

3.37 Print QR code for T500II(PrintQrCodeT500II)

[Function]

Print QR code for T500II

[Parameters]

int iSide: Units length,, that is QR code size, value is 1-9

String strData: Content

[Returned Value]

byte[] array

3.38 Print PDF417 code (PrintPdf417)

[Function]

Print PDF417 code

[Parameters]

int iDotwidth: width, value is 0-255

int iDotheight: height, value is 0-255

int iDatarows: line number

int iDatacolumns: row number

String strData: content

[Returned Value]

byte[] array

3.39 Print 1D code(Print1Dbar)

[Function]

Print 1D code

[Parameters]

int iWidth: code width, value is 2-6 units 0.125mm

int iHeight: code height, value is 1-255 units 0.125mm

int iHrsize: code show character font 0 12*24 1 9*17

int iHriseat: code show character position 0 none 1 up 2 down 3 up and down

int iCodetype: code type

```
( * UPC-A 0, * UPC-E 1, * EAN132, * EAN83,
* CODE394, * ITF5, * CODABAR6, * Standard EAN137,
* Standard EAN88, * CODE93 9, * CODE128 10)
```

String strData: code content

[Returned Value]

byte[] array

3.40 Print Disk BMP file(PrintDiskbmpfile)

[Function]

Print disk BMP file , only support single color BMP format

[Parameters]

String strPath: file path

[Returned Value]

byte[] array

3.41 Black And White Print BMP/JPG/PNG image files (PrintDiskImagefile)

[Function]

Support the picture content for the black and white print BMP/JPG/PNG files

[Parameters]

int[] pixels: image pixel array

int iWidth: image width

int iHeight: image height

[Returned Value]

byte[] array

3.42 Print BMP/JPG/PNG image files (PrintDiskImagefile)**[Function]**

Support the picture content for the black and white print BMP/JPG/PNG files

[Parameters]

String strPath: Image file path

[Returned Value]

byte[] array

3.43 Print NV BMP file(PrintNvbmp)**[Function]**

Print NV BMP file, only support single color BMP format

[Parameters]

int iNvindex: NV bit image index

int iMode: 48 normal、49 double-width、50 double-height、51 double-width、
double-height (Quadruple)

[Returned Value]

byte[] array

3.44 Get Printer Status(GetStatus)**[Function]**

Get printer status

[Parameters]

None

[Returned Value]

byte[] array

GetStatus to get the byte[] array sent to the printer, the printer to receive the return value, and then call CheckStatus, the return value is introduced into the analysis. Return value:

0 printer normal

1 printer is not connected or not on the power

2 printer and library does not match

3 print head open

4 cutting knife is not reset

5 print head overheating

6 black mark error

7 paper Exhausted

8 paper will Exhausted

Note: this interface is adapted to the serial port connection.

Support for different printer models are not the same

USB and Bluetooth interface to call GetStatus1 and GetStatus2, respectively GetStatus3, GetStatus4, GetStatus5, CheckStatus2, CheckStatus1 and the corresponding CheckStatus3,

CheckStatus4, CheckStatus5, see that the example program.

3.45 Get SDK Information(GetSDKInformation)

[Function]

Get SDK version information

[Parameters]

None

[Returned Value]

String

3.46 Converts string to an array(JNAStringToByte)

[Function]

A string of hexadecimal data converted into a byte array

[Parameters]

String strData:String

int iLen:line

[Returned Value]

byte[]

3.47 Converts array to an string (JNAByteToString)

[Function]

ByteArray to hexadecimal data string

[Parameters]

byte[] bData:

[Returned Value]

String

3.48 Converts array to an string(JNAByteToString)

[Function]

ByteArray to hexadecimal data string(Have initial position and length)

[Parameters]

byte[] bData:

int iIndex :initial position

int iLen:length

[Returned Value]

String

4. USB connection print. (UsbDirver.java)

4.1 To adapt to the development environment

USB mode connection

Android version: 3.1 and above

4.2 ConnectMethods

4.2.1 Print byte[]Array1 (write)

[Function]

Print byte[] array

[Parameters]

byte[] buf: byte[] array

[Returned Value]

int Actual print length

4.2.2 Print byte[]Array2 (write)

[Function]

Print byte[] array

[Parameters]

byte[] buf: byte[] array

UsbDevice usbDev: USB device specified

[Returned Value]

int Actual print length

4.2.3 Print byte[]Array3(write)

[Function]

Print byte[] array

[Parameters]

byte[] buf: byte[] array

int length: array length

[Returned Value]

int Actual print length

4.2.4 Print byte[]Array4(write)

[Function]

Print byte[] array

[Parameters]

byte[] buf: byte[] array

int length: array length

UsbDevice usbDev: USB device specified

[Returned Value]

int Actual print length

4.2.5 Read Data1 (read)

[Function]

Read data

[Parameters]

byte[] bytWrite: Write byte[] array

byte[] bufRead: Readbyte[] array

[Returned Value]

int Length of read data

4.2.6 Read Data2 (read)

[Function]

Read data

[Parameters]

byte[] bytWrite: Write byte[] array

byte[] bufRead: Readbyte[] array

UsbDevice usbDev: USB device specified

[Returned Value]

int Length of read data

4.2.7 USB Permission Status (isUsbPermission)

[Function]

USB Permission status

[Parameters]

None

[Returned Value]

boolean

4.2.8 Connection Status (isConnected)

[Function]

Connection status

[Parameters]

None

[Returned Value]

boolean

4.2.9 Set timeout (setFlowCtrl)

[Function]

Set USB overtime

[Parameters]

int iFlowCtrlFlag: 0:Remove USB overtime, 1:Set USB overtime

[Returned Value]

Boolean

4.2.10 Set PendingIntent(setPermissionIntent)**[Function]**

Set PendingIntent

[Parameters]

PendingIntent pi: Pending event

[Returned Value]

None

4.2.11 Connect Printer Intent1 (usbAttached)**[Function]**

Connect Printer Intent

[Parameters]

Intent intent: Specific Intent of connection

[Returned Value]

Boolean

4.2.12 Connect Printer Intent2 (usbAttached)**[Function]**

Connect Printer Intent

[Parameters]

Intent intent: Specific Intent of connection

UsbDevice usbDev: USB device specified

[Returned Value]

boolean

4.2.13 Remove Connection Intent (usbDetached)**[Function]**

Remove connection Intent

[Parameters]

Intent intent: Remove the Intent

[Returned Value]

boolean

4.2.14 Open UsbDevice1 (openUsbDevice)**[Function]**

Open UsbDevice

[Parameters]

None

[Returned Value]

Boolean

4.2.15 Open UsbDevice2 (openUsbDevice)

[Function]

Open UsbDevice

[Parameters]

UsbDevice usbDev: USB device specified

[Returned Value]

boolean

4.2.16 Close UsbDevice1 (closeUsbDevice)

[Function]

Close UsbDevice(Not parameter)

[Parameters]

None

[Returned Value]

None

4.2.17 Close UsbDevice2 (closeUsbDevice)

[Function]

Close UsbDevice(Have parameter)

[Parameters]

UsbDevice usbDev: USB device specified

[Returned Value]

None

4.3 Sample

See `UsbPrinterUsb` project for details.

Compiled under Android Studio 3.4.1.

Manifest file Activity to join the action insert Usb

```
<intent-filter>
```

```
    <action android:name= "android.hardware.usb.action.USB_DEVICE_ATTACHED" />
```

```
</intent-filter>
```

```
<meta-data
```

```
    android:name= "android.hardware.usb.action.USB_DEVICE_ATTACHED"
```

```
    android:resource= "@xml/device_filter" />
```

`@xml/device_filter`, Add manufacturer ID and product ID filter, when this model equipment

through the Usb connected to the system, the corresponding Activity will start

```
<resources>
```

```
    <usb-device vendor-id= "1027" product-id= "24592" />
```

```
    <usb-device vendor-id= "1027" product-id= "24596" />
```

```
</resources>
```

```
//init
```

```
public void onCreate(Bundle savedInstanceState) {  
  
    mUsbDriver = new com.printsdk.usbsdk.UsbDriver(  
        (UsbManager) getSystemService(Context.USB_SERVICE), this);  
  
    PendingIntent permissionIntent = PendingIntent.getBroadcast(this, 0,  
        new Intent("com.usb.sample.USB_PERMISSION"), 0);  
  
    mUsbDriver.setPermissionIntent(permissionIntent);  
  
    // Broadcast listen for new devices  
  
    IntentFilter filter = new IntentFilter();  
  
    filter.addAction(UsbManager.ACTION_USB_DEVICE_ATTACHED);  
  
    filter.addAction(UsbManager.ACTION_USB_DEVICE_DETACHED);  
  
    this.registerReceiver(mUsbReceiver, filter);  
  
}
```

```
// BroadcastReceiver when insert/remove the device USB plug into/from a USB
```

```
BroadcastReceiver mUsbReceiver = new BroadcastReceiver() {  
    public void onReceive(Context context, Intent intent) {  
        String action = intent.getAction();  
        if (UsbManager.ACTION_USB_DEVICE_ATTACHED.equals(action)) {  
            UsbDevice device = (UsbDevice) intent  
                .getParcelableExtra(UsbManager.EXTRA_DEVICE);  
            if ((device.getProductId()==8211 && device.getVendorId()==1305)  
                || (device.getProductId()==8213 &&
```

```

        device.getVendorId() == 1305))
    {
        if (mUsbDriver.usbAttached(intent)) {
            mUsbDriver.openUsbDevice(device);
        }
    }
} else if (UsbManager.ACTION_USB_DEVICE_DETACHED.equals(action)) {
    UsbDevice device = (UsbDevice) intent
        .getParcelableExtra(UsbManager.EXTRA_DEVICE);
    if ((device.getProductId() == 8211 && device.getVendorId() == 1305)
        || (device.getProductId() == 8213 &&
            device.getVendorId() == 1305))
    {
        mUsbDriver.closeUsbDevice(device);
    }
}
};

//print
mUsbDriver.write(byte[]);

```

4.4 Matters Needing Attention

Under normal circumstances, when running the APP connection USB printer, will appear to allow the application to access the USB?. If you do not have the prompt box can not be normal use, generally for the Android system permissions.

Can refer to the following information:

into folder

/system/etc/permissions

in that folder find file named

handheld_core_hardware.xml or tablet_core_hardware.xml

and add

```
<feature name="android.hardware.usb.host">
```

into <permissions> section

Reboot your device. Usb host api should work.

5. Bluetooth Connection Print

5.1 To adapt to the development environment

Bluetooth mode connection

Android version: 2.2 and above

5.2 **ConnectMethods** (BluetoothManager.java)

5.2.1 **Connect Bluetooth Server(connectServer)**

[Function]

Connect Bluetooth server

[Parameters]

None

[Returned Value]

BluetoothSocket

5.2.2 **Send Data or Instruction (send)**

[Function]

Send data or instruction

[Parameters]

byte[] bSend: Data or instruction array that needs to be sent

[Returned Value]

None

5.2.3 Send Data or Instruction in batches (send)

[Function]

Send data or instruction

[Parameters]

byte[] bSend: Data or instruction array that needs to be sent

int iModlen: The length of each send

int iSleep: The delay in milliseconds between each transmission

[Returned Value]

None

5.2.4 Read Data (read)

[Function]

Read data

[Parameters]

byte[] bSend: Data or instruction array that needs to be sent

[Returned Value]

byte[]

5.2.5 Shutdown Client (shutdownClient)

[Function]

Shutdown client

[Parameters]

None

[Returned Value]

None

5.2.6 Whether The Socket Connection (isConnected)

[Function]

To determine whether the socket connection

[Parameters]

None

[Returned Value]

boolean

5.3 Sample

See UsbPrinterBt project for details.

Compiled under Android Studio 3.4.1.

6. Serial Connection Print

6.1 To adapt to the development environment

Serial connection mode

6.2 Sample

6.2.1 Specify device name(SetDevname)

[Function]

Open serial port printing

[Returned Value]

int iDevtype:Printer device connection mode, Fixed to 1

String cDevname:Printer port name

int iBaudrate:Printer baud rate

[Returned Value]

int (0:succeed, 1:fail)

6.2.2 Init printer(SetInit)

[Function]

Init printer

[Returned Value]

None

[Returned Value]

int (0:succeed, 1:fail)

6. 2. 3 Close device(SetClose)

[Function]

Close device

[Parameters]

None

[Returned Value]

int (0:succeed, 1:fail)

6. 2. 4 PrintData(PrintTransmitJNA)

[Function]

Print data

[Parameters]

String strCmd:String data, Hexadecimal string format

int iLength:String length

[Returned Value]

int (0:succeed, 1:fail)

6. 2. 5 Read data(GetTransmitJNA)

[Function]

Read the data returned by the printer

[Parameters]

String strCmd:String data, Hexadecimal string format

int iLength:String length

[Returned Value]

int (0:succeed, 1:fail)

6.3 Sample

See the `UsbPrinterCom` project for details.

Compiled under Android Studio 3.4.1.

6.4 Matters needing attention

The printer needs to call the `SetDevname` parameter before initializing the `SetInit()` interface. If not called, the `SetInit()` method will fail to run and crash

The printer should be initialized before using it, and the `SetInit()` interface should not be called frequently.